

## SPLIT-KEY KEY-AGREEMENT PROTOCOL

The present invention relates to the field of key agreement protocols in cryptographic systems.

5

### BACKGROUND OF THE INVENTION

Traditionally, entities communicated on paper and were able to ensure privacy in many ways. The transition from paper to electronic media however, has created the need for electronic privacy and authenticity. In cryptographic schemes, the entities use primitives, which are mathematical operations together with encoding and formatting techniques to provide security. For each scheme the parties participating in the scheme normally agree upon or exchange certain information before executing the scheme function. The specific information that needs to be agreed upon is detailed for each scheme. Such agreement may be achieved by any means suitable for the application. It may be implicitly built into the system or explicitly achieved by some sort of exchange of information with or without involvement from other parties. In particular, parties often need to agree on parameters and obtain each other's public keys. For proper security, a party needs to be assured of the true owners of the keys and parameters and of their validity. Generation of parameters and keys needs to be performed properly and, in some cases, verification needs to be performed.

10

15

20

In general, the different types of schemes may be defined as follows. Key agreement schemes, in which two parties use their public, private key pairs and possibly other information, to agree on a shared secret key. A signature scheme with appendix is a scheme in which one party signs a message using its private key and any other party can verify the signature by examining the message, the signature, and the signer's cross corresponding public key. In signature schemes with message recovery, one party signs a message using its private key and any other party can verify the signature and recover the message by examining the signature and the signer's corresponding public key. Finally, in encryption schemes, any party can encrypt a message using the recipient's public key and only the recipient can decrypt the message using its corresponding private key.

25

30

An example of a key derivation scheme is the MQV (Menezes-Qu-Vanstone). In the MQV scheme, a shared secret value is derived from one party's two key pairs and another

party's two public keys where all the keys have the same discrete log (DL) parameters. In this generalized MQV scheme, it is assumed that the shared secret value is that which is shared between two parties.

However, where each party or entity consists of a collection of parties say  $A = \{A_1, A_2, \dots, A_n\}$  and  $B = \{B_1, B_2, \dots, B_m\}$  where  $m$  is not necessarily equal to  $n$  and at least one of  $m$  or  $n$  is at least two (that is, not both  $A$  and  $B$  consist of one individual), it is difficult to implement the generalized MQV scheme if these two entities wish to establish a common key in order to communicate privately.

## 10 SUMMARY OF THE INVENTION

Accordingly, the present invention seeks to provide a solution to the problem of establishing a common key for private communication between entities wherein the entities include a collection of sub entities.

An advantage of the present invention is that all members of each entity must participate in the scheme and no subcollection of either entity can impersonate its entire entity.

In accordance with this invention there is provided a method for generating a shared secret value between entities in a data communication system, one or more of the entities having a plurality of members for participation in the communication system, each member having a long term private key and a corresponding long term public key, the method comprising the steps of:

- (a) generating an entity long term private key and corresponding entity long term public key for each entity by combining the long term private and public keys of each members of the entity.
- (b) generating a short term private and a corresponding short term public key for each of the members;
- (c) exchanging short term public keys of the members within an entity;
- (d) for each member:
  - i. computing an intra-entity shared key by mathematically combining said short term public keys of each said member;
  - ii. computing an intra-entity public key by mathematically combining its short - term private key, the long term private key and said intra-entity shared key;

- (e) for each entity combining intra-entity public keys to derive a group short-term public key;
- (f) each entity transmitting its intra-entity shared key and its group short term public key to said other entities; and
- 5 (g) each entity computing a common shared key K by combining its group short term public key, with the intra-entity shared key, and an entity long term public key received from the other entity.

## 10 BRIEF DESCRIPTION OF THE DRAWINGS

A preferred embodiment of the invention will now be described by way of example only with reference to the accompany drawings in which:

Figure 1 is a schematic diagram of a communication system; and

Figure 2 is a schematic diagram illustrating the steps of a protocol to establish a common key.

## 15 DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Referring to figure 1, a schematic diagram of a communication system is shown generally by numeral 10. The system 10 includes a first entity A (12) and a second entity B (14) that exchange data over a communication channel 16. Each of the entities A and B include members  $A_1, A_2 \dots A_n$ , and  $B_1, B_2 \dots B_n$ , respectively. For convenience, the embodiment described has two members  $A_1, A_2$  and  $B_1, B_2$  although it will be appreciated that typically each entity will have several members. It is assumed the entities A and B include processors for performing cryptographic operations and the like. The members  $A_1, A_2$  may for example be a first group of users on a local area network (LAN) that wish to communicate securely with a second group of users  $B_1, B_2$  on a second LAN or even on the same LAN. In either case the computations may be performed for the entities A (12) and B (14) by for example a LAN server 18 or the like, provided that each member has its own secure boundary.

Each entity and its associated members  $A_i, B_i$  have been initialized with the same system parameters. The system parameters for this exemplary protocol are an elliptic curve point P, which is the generating point of an elliptic curve over  $F_2^m$  of order  $n$ . Additionally, each of the members is initialized with respective long-term public and private key pairs. That is, each of

the members  $A_i$  has long term private and public key pairs  $(a_i, a_iP)$  and each of the member  $B_i$  have long term private and public key pairs  $(b_i, b_iP)$ , respectively.

Each of the entities A, B generates respective long-term public keys derived from the long-term public keys of each of its members. The long-term private key  $a$  of the entity A is then  $(a_1 + a_2 + \dots a_n)$  and its corresponding long-term public key,  $aP$ , is  $(a_1 + a_2 + \dots a_n)P$ . In the present example the key pair  $(a, aP)$  of entity A is  $(a_1 + a_2); (a_1 + a_2)P$ . Similarly, for entity B its long-term private key  $b$  is  $(b_1 + b_2)$  and its corresponding long-term public key is  $bP (b_1 + b_2)P$ . The entity long-term public keys  $aP, bP$  can be computed by summing the members public keys. The entity public keys are published by the respective entities, and if appropriate certified by a trusted authority or CA trusted by all of the entities.

Typically, entities A (12) and B (14) wish to agree upon a common key, which may then be used for subsequent cryptographic communications between the activities.

Referring thus to figure 2, a schematic diagram of an embodiment of a suitable protocol is shown generally by numeral 40. The member  $A_1$  generates a random value  $x_1$  (its short-term private key, also known as ephemeral or session key) and computes a corresponding value  $x_1P$  (its short-term public key); similarly, member  $A_2$  generates a random value  $x_2$  and computes a corresponding value  $x_2P$ . Preferably  $0 < a_i < n-1$  and  $0 < x_i < n-1$ . Next, the members of the entity A exchange their session public keys  $x_iP$ . In the present example,  $A_2$  and  $A_1$  exchange their session public keys  $x_1P$  and  $x_2P$  denoted  $X_1$  and  $X_2$  respectively. This may be termed a first intra-entity key exchange.

Next, member  $A_1$  computes  $r = x_1P + x_2P$  and similarly, ~~entity~~ <sup>member</sup>  $A_2$  computes  $r = x_2P + x_1P$ . Thus, establishes an intra-entity shared key available and containing a contribution from each member of the entity.

The entity A transmits the intra-entity shared key  $r$  to the entity B with whom it wishes to establish a common key  $K$ .

Next, member  $A_1$  computes a short term intra-entity public key  $s_1$  using its short term private key and long term private key combined with a function  $f$  of the intra-entity public key, that is  $s_1 = x_1 + a_1 f(r) \pmod n$ , where  $f$  is typically a hash function such as SHA-1 and  $n$  is the order of the curve. Similarly, member  $A_2$  computes its intra-entity public key  $s_2 = x_2 + a_2 f(r) \pmod n$ .

The entity A computes an entity or group short term public key, which is derived from a summation of the intra-entity public key of each member  $s = s_1 + s_2 = x_1 + x_2 + (a_1 + a_2) f(r) \bmod (n)$ .

5 The entity B similarly computes the analogous information using its own public and private keys using the same computations performed by entity A. Thus, each member of B computes an intra-entity shared key  $\bar{r}$  using the short term public keys of each of the members and  $\bar{r}$  is forwarded to entity A. Next, each of the members in B compute their own intra-entity public key  $t_i = y_i + b_i f(\bar{r}) \bmod (n)$  and computes the group short-term public key  $t = t_1 + t_2$ .

The entity A then computes a value K which is the shared key between the entities A and  
10 B by retrieving the long term public key, bP, of entity B and computing  $K = s(\bar{r} + (bP) f(\bar{r})) = s(t)P$ . The entity B also retrieves the long term public key aP of entity A and computes K using t, r, and aP, i.e.  $K = t(r + aP.f(r)) = t(s)P$ .

Consequently, if a member of the entity A, either  $A_1$  or  $A_2$ , is not present in the scheme then the group short term public key, s, changes, as does the value for K. Therefore,  
15 communication with entity B would not be successful without establishing a new session. Similarly, if either  $B_1$  or  $B_2$  is not present in the scheme then the group short term public key, t, changes, altering the value of K. In this case, communication with A would not be successful without establishing a new session.

Accordingly, the present protocol ensures that all members of each entity must participate  
20 in the scheme and no sub-collection of either entity can impersonate its entire entity.

Although the above scheme has been described with respect to the elliptic curve systems which is an additive group, it may analogously be used in multiplicative groups. Furthermore the above protocol although exemplified with two members per entity, may be generalized where each party or entity consists of a collection of members say  $A = \{A_1, A_2 \dots A_n\}$  and  $B = \{B_1, B_2, \dots B_m\}$  where m is not necessarily equal to n and at least one of m or n is at least two (that is, not  
25 both A and B consist of one individual).

Although the invention has been described with reference to certain specific embodiments, various modifications thereof will be apparent to those skilled in the art without departing from the spirit and scope of the invention as outlined in the claims appended hereto.